

ICS 第三次实验实验报告

李浩然 2025 年 11 月 7 日

1 实验目的

1. 掌握 LC-3 汇编语言的基本语法和编程规范，熟练运用数据传送、算术运算、分支跳转等核心指令。
2. 实现基于模 7 运算的简易数据校验算法。
3. 学会处理输入输出数据的 ASCII 码转换，掌握循环结构和条件判断在汇编程序中的实现方法。
4. 培养汇编程序的调试能力，能够定位并解决程序运行过程中的逻辑错误。

2 实验原理

2.1 核心算法逻辑

1. 初始化：设定 ASCII 码转换基准值、终止符标识和模值常量，初始化结果寄存器。
2. 输入处理：循环读取输入字符，若检测到终止符'y' 则停止输入。
3. 数据转换：将输入数字的 ASCII 码转换为对应的数值。
4. 校验运算：对当前结果执行左移 1 位（等价于乘 2）操作，累加新输入的数值，若结果大于等于 7 则执行减 7 的模运算。
5. 结果输出：将最终校验结果转换为 ASCII 码并输出。

2.2 ASCII 码转换原理

数字字符的 ASCII 码值与对应数值相差 48（即十六进制 x0030），通过将输入字符的 ASCII 码减去 48 可得到对应的数值；反之，将数值加上 48 可转换为对应的 ASCII 码用于输出。

3 实验内容与步骤

3.1 程序编写

根据实验原理编写 LC-3 汇编程序，实现上述校验算法，程序代码如下：

```

1      .ORIG x3000
2
3      LD    R3, ASCII0      ; R3='0' (48), 用于ASCII码与数值的转换基准
4      LD    R4, ASCIIy      ; R4='y' (121), 输入终止符
5      LD    R5, SEVEN       ; R5=7, 模运算的基数
6
7      AND   R1, R1, #0      ; R1初始化, 用于存储校验结果
8
9  READ_LOOP
10     GETC                    ; 读取输入字符, 存入R0
11
12     ; 检测是否为终止符'y'
13     NOT   R6, R4
14     ADD   R6, R6, #1      ; 计算R4的补码, 即-'y'
15     ADD   R6, R0, R6      ; R6 = 输入字符 - 'y', 结果为0则表示输入'y'
16     BRz   OUTPUT_RESULT   ; 若为终止符, 跳转到结果输出环节
17
18     ; ASCII码转换为对应数字
19     NOT   R7, R3
20     ADD   R7, R7, #1      ; 计算R3的补码, 即-'0'
21     ADD   R2, R0, R7      ; R2 = 输入字符 - '0', 得到对应数字值
22
23     ; 校验结果乘2 (左移1位)
24     ADD   R1, R1, R1
25
26     ; 累加当前输入的数字
27     ADD   R1, R1, R2
28
29     ; 执行模7运算
30     NOT   R6, R5
31     ADD   R6, R6, #1      ; 计算R5的补码, 即-7
32     ADD   R6, R1, R6      ; R6 = 当前结果 - 7
33     BRn   SKIP_SUB7      ; 若结果小于7, 跳过减7操作
34     ADD   R1, R6, #0      ; 若结果大于等于7, 执行减7更新结果
35
36  SKIP_SUB7
37     BR    READ_LOOP      ; 跳回循环, 继续读取下一个字符
38
39  OUTPUT_RESULT
40     ; 将校验结果转换为ASCII码并输出
41     ADD   R0, R1, R3
42     OUT
43     HALT
44
45 ; 数据定义
46 ASCII0  .FILL x0030      ; 0的ASCII码值
47 ASCIIy  .FILL x0079      ; 'y'的ASCII码值
48 SEVEN   .FILL #7

```

49

50

.END

4 实验结果与分析

4.1 结果分析

- 程序能够准确识别终止符'y'，停止输入处理并输出结果，符合设计需求。
- 模 7 运算模块工作正常，当结果大于等于 7 时能正确执行减 7 操作，确保结果在 0-6 范围内。
- ASCII 码转换功能稳定，实现了数字字符与数值之间的正确转换。

5 实验总结与体会

5.1 实验总结

本次实验成功实现了基于模 7 运算的 LC-3 程序，通过 LC-3 汇编语言完成了输入处理、数据转换、循环运算和结果输出等功能。程序经多组测试验证，实际运行逻辑正确，能够准确完成数据校验任务。在实验过程中，深入理解了汇编语言的指令执行机制，掌握了循环结构和条件分支的实现方法。

5.2 问题与解决方法

编写时模运算模块逻辑错误，未正确处理补码计算。通过调试跟踪寄存器值，修正了补码运算的指令顺序，确保减 7 操作的正确性。